

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **06103225 A**

(43) Date of publication of application: 15.04.94

(51) Int. Cl.

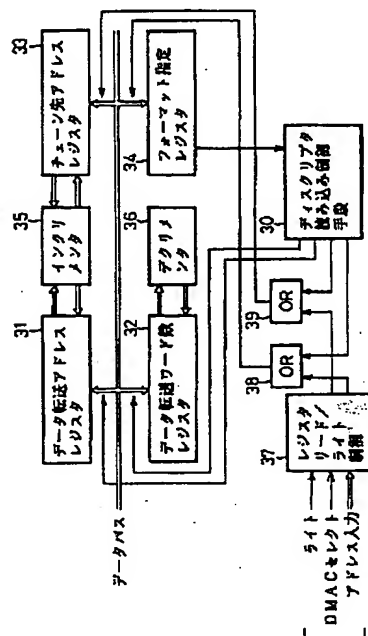
G06F 13/28(21) Application number: **04249267**(22) Date of filing: **18.09.92**(71) Applicant: **FUJITSU LTD FUJITSU DEVICE KK**(72) Inventor: **TARUI YUJI
MIKAMI GOJI**(54) **CHAIN TYPE DMA SYSTEM AND DMA CONTROLLER THEREFOR**

(57) Abstract:

PURPOSE: To execute flexible chaining without expanding a discripiter area and increasing the reading operation of a discripiter.

CONSTITUTION: In a chain type DMA controller provided with a discripiter information reading control means 30 and a register group 31 to 33 for storing discripiter information, the register group is provided with a format specifying register 34 for storing format specifying information for regulating the format of succeeding discripiter information and the control means 30 is constituted so as to read out the succeeding discripiter information in accordance without the format specifying information.

COPYRIGHT: (C)1994,JPO&Japio



(11)特許出願公開番号

特開平6-103225

(43)公開日 平成6年(1994)4月15日

技術表示箇所

9072-5B

審査請求 未請求 請求項の数 6 (全 20 頁)

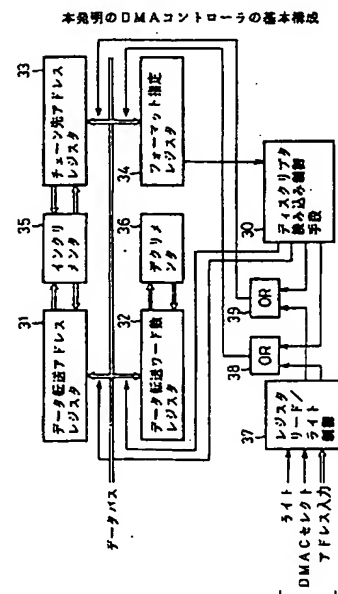
(74)代理人 弁理士 青木 朗 (外3名)

(54)【発明の名称】 チェーン式DMA方式及びそのためのDMAコントローラ

(57) 【要約】

【目的】 本発明はチェーン式DMA転送方式及びそのためのDMAコントローラに関し、ディスクリプタ領域を大きくせず、ディスクリプタの読み込み動作量も増大させずに柔軟なチェーンが行なえるようにすることを目的とする。

【構成】 ディスクリプタ情報読み込み制御手段30と、ディスクリプタ情報を記憶するレジスタ群31、32、33とを備えるチェーン式DMAコントローラであって、レジスタ群は次のディスクリプタ情報の形式を規定するフォーマット指定情報を記憶するフォーマット指定レジスタ34を備え、ディスクリプタ読み込み制御手段30は、フォーマット指定情報に従って次のディスクリプタ情報を読み込むように構成する。



【特許請求の範囲】

【請求項1】 外部メモリのディスクリプタ領域に格納されているディスクリプタ情報を読み込み、当該ディスクリプタ情報に従ってDMAデータ転送を行なう一連の動作を前記ディスクリプタ情報を逐次読み込みながら繰り返し行なうチェーン式DMA方式であって、前記ディスクリプタ情報は、次のディスクリプタ情報の形式を規定するフォーマット指定情報を有し、最初のディスクリプタ情報の読み込み以外は、読み込まれた前記フォーマット指定情報に従って次のディスクリプタ情報の読み込みを行なうことを特徴とするチェーン式DMA方式。

【請求項2】 外部メモリのディスクリプタ領域に格納されているディスクリプタ情報を読み込み、当該ディスクリプタ情報に従ってDMAデータ転送を行なう一連の動作を前記ディスクリプタ情報を逐次読み込みながら繰り返し行なうチェーン式DMA方式であって、前記ディスクリプタ情報は、当該ディスクリプタ情報の形式を規定するフォーマット指定情報を所定位置に有し、前記ディスクリプタ情報の読み込み時には、前記フォーマット指定情報を含む共通部分をまず読み取り、読み込んだ前記フォーマット指定情報に従って残りの部分を読み込むことを特徴とするチェーン式DMA方式。

【請求項3】 請求項1又は2のいずれかに記載のチェーン式DMA方式であって、前記DMAデータ転送後に転送状態等の状態情報を前記外部メモリのディスクリプタ領域上に書き出すチェーン式DMA方式において、前記状態情報の書き出しは、読み込んだ前記フォーマット指定情報に従って行なうことを特徴とするチェーン式DMA方式。

【請求項4】 外部メモリのディスクリプタ領域に格納されているディスクリプタ情報の読み込みを制御するディスクリプタ情報読み込み制御手段(30)と、読み込んだ前記ディスクリプタ情報を記憶するレジスタ群(31, 32, 33)とを備え、前記レジスタ群(31, 32, 33)に記憶されたディスクリプタ情報に従ってDMAデータ転送を行なう一連の動作を前記ディスクリプタ情報を逐次読み込みながら繰り返し行なうチェーン式DMAコントローラであって、前記レジスタ群は、読み込まれた前記ディスクリプタ情報に含まれる次のディスクリプタ情報の形式を規定するフォーマット指定情報を記憶するフォーマット指定レジスタ(34)を備え、前記ディスクリプタ読み込み制御手段(30)は、前記フォーマット指定レジスタ(34)に記憶されたフォーマット指定情報に従って次のディスクリプタ情報の読み込みを行なうことを特徴とするチェーン式DMAコントローラ。

【請求項5】 外部メモリのディスクリプタ領域に格納されているディスクリプタ情報の読み込みを制御するデ

ィスクリプタ情報読み込み制御手段(30)と、読み込んだ前記ディスクリプタ情報を記憶するレジスタ群(31, 32, 33)とを備え、前記レジスタ群(31, 32, 33)に記憶されたディスクリプタ情報に従ってDMAデータ転送を行なう一連の動作を前記ディスクリプタ情報を逐次読み込みながら繰り返し行なうチェーン式DMAコントローラであって、

前記レジスタ群は、読み込まれた前記ディスクリプタ情報の所定位置に含まれる当該ディスクリプタ情報の形式を規定するフォーマット指定情報を記憶する前記フォーマット指定レジスタ(34)を備え、前記ディスクリプタ読み込み制御手段(30)は、前記フォーマット指定情報を含む共通部分をまず読み取って前記レジスタ群の対応部分に記憶し、前記フォーマット指定レジスタ(34)に記憶されたフォーマット指定情報に従って残りの部分を読み込むことを特徴とするチェーン式DMAコントローラ。

【請求項6】 請求項4又は5のいずれかに記載のチェーン式DMAコントローラであって、前記DMAデータ転送後に転送状態等の状態情報を前記外部メモリのディスクリプタ領域上に書き出す書き出し手段を備え、該書き出し手段は、前記フォーマット指定レジスタ(34)に記憶されたフォーマット指定情報に従って前記状態情報を書き出すことを特徴とするチェーン式DMAコントローラ。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、データブロックを高速に転送するためのDMA方式及びそのためのDMAコントローラに関し、特に一度の起動で複数のブロックを異なるアドレスとの間で転送できるチェーン式DMA方式及びそのためのDMAコントローラに関する。

【0002】

【従来の技術】 データブロックを高速に転送するためにDMAデータ転送が広く用いられている。初期のDMA方式は、中央処理装置(CPU)がDMAコントローラに転送先(又は転送元)のアドレスと転送するデータ量(バイト数又はワード数)を書き込んだ上で、DMAコントローラに起動コマンドを発行することでDMAデータ転送を開始していた。書き込まれた転送データ量の転送が終了すると、その転送動作が終了する。従って転送動作毎に転送先アドレス等の動作情報を書き込む必要があった。

【0003】 近年のコンピュータシステムの高速化、高効率化要求に伴い、従来から高速なデータブロックの転送に用いられていたDMAコントローラにも、一層効率的なデータ転送を行うことが要求されてきた。この為、DMAコントローラに対してデータ転送先アドレス、データ転送先メモリのワード数を複数ブロック分あらかじめ設定しておくことができる、チェーン式のDMAコン

トローラが提供されている。

【0004】チェーン式DMAコントローラは、外部メモリのディスクリプタ領域と呼ばれる部分に格納されている転送先アドレス及び転送データ量（以下の説明においては転送データ量がワード単位で表わされるものとして、転送ワード数と称することにする。）等の転送動作に必要な情報を順に読み込み、各情報を読み込む度に、その情報の内容に応じた転送動作を行なう。ディスクリプタ領域に格納された情報をディスクリプタ情報、又は単にディスクリプタと称する。

【0005】従来のチェーン方式のDMAコントローラは、DMAコントローラに対するモード設定により、複数種類の形式のディスクリプタを取り扱うことができる。ディスクリプタの形式として代表的なのは、アレイチェーンモードとリンクアレイチェーンモードである。この他にも各種のディスクリプタがあり得るが、ここではこの2つのモードを例として説明を行なう。

【0006】アレイチェーンモードに設定しておく、1つのディスクリプタはデータ転送アドレスとデータ転送ワード数で構成され、CPUとDMAコントローラは図13のフローチャートの様に動作する。

(1) ステップ1301で、CPUは、図14に示すディスクリプタテーブルを用意し、所定のメモリ位置に書き込む。

【0007】(2) ステップ1302で、CPUは、ディスクリプタ領域1が格納されているアドレスとアレイチェーンのフォーマットをDMAコントローラに書き込む。これに応じてDMAコントローラ(DMAC)は、ディスクリプタ情報の開始アドレスとアレイチェーンフォーマットの設定を行なう。

(3) ステップ1303で、CPUは、DMACに起動コマンドを発行し、ステップ1312でDMACが起動する。

【0008】(4) ステップ1313で、DMACは、CPUが設定したフォーマットに従ってディスクリプタ領域1に格納されている内容を読み込む。

(5) ステップ1314で、すべてのディスクリプタ情報を読み込んだかを判定し、読み込みが終了していなければ、ステップ1318でディスクリプタ情報のアドレスを増加させる。

【0009】(6) ステップ1315と1316で、DMACは、ディスクリプタ領域1に示されていたデータ転送アドレスからディスクリプタ領域1に示されていたデータ転送ワード数分のDMAデータ転送を行う。

(7) DMACは、ステップ1317で最終ディスクリプタであるか判定し、最終でなければ、ステップ1318でディスクリプタ情報のアドレスを増加させ、ステップ1313に戻る。そしてディスクリプタ領域1と連続しているディスクリプタ領域2に格納されている内容を読み込む。

【0010】(8) 上記と同様にDMACは、ディスクリプタ領域2に示されていたデータ転送アドレスからディスクリプタ領域2に示されていたデータ転送ワード数分のDMAデータ転送を行う。

(9) 以降、上記の動作を、ディスクリプタ領域nに格納されている内容に従ったDMAデータ転送までを行う。そして最終ディスクリプタであればステップ1317で待機状態に戻る。

【0011】なお図14はアレイチェーンモード時に必要とする最小構成のディスクリプタを示したものであり、実際には他の情報が含まれることがある。しかしここでは省略している。これは以下の説明でも同様である。リンクアレイチェーンモードに設定しておく、1つのディスクリプタはデータ転送アドレスとデータ転送ワード数とチェーン先アドレスで構成され、CPUとDMAコントローラは図15のフローチャートの様に動作する。

【0012】(1) ステップ1501でCPUは、図16に示すディスクリプタテーブルを用意し、所定のメモリ位置に書き込む。

(2) ステップ1502でCPUは、ディスクリプタ領域1が格納されているアドレスとリンクアレイチェーンのフォーマットをDMACに書き込む。これに応じてDMACはディスクリプタ情報の開始アドレスとリンクアレイチェーンフォーマットの設定を行なう。

【0013】(3) ステップ1503でCPUは、DMAコントローラに起動コマンドを発行する。これに応じてステップ1512で、DMACが起動する。

(4) ステップ1513で、DMACは、CPUが設定したディスクリプタ領域1に格納されている内容を読み込む。

(5) ステップ1514で、すべてのディスクリプタ情報を読み込んだかを判定し、読み込みが終了していなければ、ステップ1519でディスクリプタ情報のアドレスを増加させる。

【0014】(6) ステップ1515と1516で、DMACは、ディスクリプタ領域1に示されていたデータ転送アドレスからディスクリプタ領域1に示されていたデータ転送ワード数分のDMAデータ転送を行う。

(7) DMACは、ステップ1517で最終ディスクリプタであるか判定し、最終でなければ、ステップ1518でディスクリプタ情報のアドレスを増加させ、ステップ1513に戻る。そしてディスクリプタ領域1に示されていたチェーン先アドレス上のディスクリプタ領域2に格納されている内容を読み込む。

【0015】(8) そして上記と同様にDMACは、ディスクリプタ領域2に示されていたデータ転送アドレスからディスクリプタ領域2に示されていた転送ワード数分のDMAデータ転送を行う。

(9) 以降、上記の動作を、ディスクリプタ領域nに格

納されている内容に従ったDMAデータ転送までを行う。そして最終ディスクリプタの時にはステップ1517で待機状態に戻る。

【0016】リンクアレイチェーンモードは、アレイチェーンモードの様に、各ディスクリプタを連続して格納しておく必要がなく、外部メモリへのディスクリプタの書き込みはより柔軟に行なえる。しかしリンクアレイチェーンモードでは、ディスクリプタがチェーン先アドレスを含むため、その読み込みに余分の時間を要する。図17は従来のDMAコントローラのブロック構成図である。

【0017】図17において、150はディスクリプタの読み込みを制御する制御部である。151から154は、データ転送アドレスレジスタ、データ転送ワード数レジスタ、チェーン先アドレスレジスタ及びフォーマット制御レジスタである。155はデータ転送アドレスレジスタのデータを増加させるインクリメンタであり、アレイチェーンモード時はチェーン先アドレスレジスタのインクリメントにも使用される。156はデータ転送時に1回データ転送を行なう度にデータ転送ワード数レジスタのデータを減少させるデクリメンタであり、データ転送ワード数レジスタのデータがゼロになればそのデータ転送が終了する。157はCPUからの信号に応じてチェーン先アドレスレジスタ153とフォーマット制御レジスタ154への書き込み等を行なう部分である。158はチェーン先アドレスレジスタ153への書き込みデータを選択するためのORゲートである。

【0018】ここで注目すべき点は、アレイチェーンモードであるかリンクアレイチェーンモードであるかは開始前にCPUからDMAコントローラ(DMAC)に書き込まれ、フォーマット制御レジスタ154に記憶されることである。設定されたモードはそのDMA転送動作中変化しない点である。

【0019】

【発明が解決しようとする課題】 前述のように、アレイチェーンモードとリンクアレイチェーンモードにはそれぞれ利点があり、必要に応じて使い分けられていた。しかしそれぞれのモードには、次のような問題点があった。従来のアレイチェーンモードでは、チェーンをループ状に構成することが出来ず、起動したDMACがチェーン最終までのDMA転送を終了した後、再びCPUがDMACにディスクリプタ領域1が格納されているのアドレスを設定し直す必要が生じる。一連のDMA転送動作の前に、CPUがこのような設定動作を行なうのは煩雑であるという問題がある。

【0020】一方、リンクアレイチェーンモードでは、最終ディスクリプタ領域のチェーン先アドレスを最初のディスクリプタ領域のアドレスとすることで、チェーンをループ状に構成することができる。しかし、前述のように、リンクアレイチェーンモードでは、各ディスクリ

プタ領域にチェーン先アドレスが含まれている為、毎回チェーン先アドレスを読み込むバスサイクルが必要になってしまうという問題がある。

【0021】本発明は、上記問題点に鑑みてなされたものであり、起動されたDMAコントローラが、チェーン途中であってもディスクリプタフォーマットを切り換え、ディスクリプタアクセス回数の削減、ディスクリプタ領域の削減等を行える柔軟性のあるDMA方式及びそのためのDMAコントローラを提供することを目的とする。

【0022】

【課題を解決するための手段】 図1は本発明におけるディスクリプタ情報のフォーマット毎の基本構成を示す図であり、図2は図1のフォーマットのディスクリプタを組み合わせたディスクリプタ全体構成例を示す図であり、図3は本発明のDMAコントローラの基本構成を示す図であり、図4は本発明におけるDMA転送動作の基本的な処理を示す図である。

【0023】本発明のチェーン式DMA方式は、上記目的を達成するため、図1及び図2に示すように各ディスクリプタ情報が次のディスクリプタ情報の形式を規定するフォーマット指定情報を有するようにし、最初のディスクリプタ情報の読み込み以外は、読み込まれたフォーマット情報に従って次のディスクリプタ情報の読み込みを行なうように構成する。

【0024】また本発明のチェーン式DMAコントローラは、外部メモリのディスクリプタ領域に格納されているディスクリプタ情報の読み込みを制御するディスクリプタ読み込み制御手段30と、読み込んだディスクリプタ情報を記憶するレジスタ群31、32、33とを備えており、レジスタ群31、32、33に記憶されたディスクリプタ情報に従ってDMAデータ転送を行なう一連の動作を繰り返し行なうが、上記目的を達成するため、レジスタ群は、読み込まれたディスクリプタ情報に含まれる次のディスクリプタ情報の形式を規定するフォーマット情報を記憶するフォーマット指定レジスタ34を備え、ディスクリプタ読み込み制御手段30はフォーマット指定レジスタ34に記憶されたフォーマット指定情報に従って次のディスクリプタ情報の読み込みを行なうように構成する。

【0025】

【作用】 図1及び図2に示すように、本発明のディスクリプタ情報にはフォーマット指定情報が含まれている、第1の態様では、フォーマット指定情報は次のディスクリプタ情報の形式を規定しており、このフォーマット指定情報に従って次のディスクリプタ情報の読み込みが行なわれる。従って、チェーン式のDMA転送において、途中でディスクリプタ情報のフォーマットを変更することが可能になる。

【0026】 途中でのディスクリプタ情報のフォーマッ

トの変更を可能にすることにより、効率的でより柔軟な動作が可能になる。図3及び図4を参照して、本発明のDMAコントローラ(DMAC)の基本構成とその基板動作を説明する。なお、ディスクリプタ情報は図1及び図2の構成を有しているものとする。

【0027】図3において、30はディスクリプタ読み込み制御手段であり、ディスクリプタ情報を各レジスタに読み込むための制御信号を出力する。31はデータ転送アドレスレジスタであり、転送先又は転送元(以下転送先として説明する。)のメモリのアドレスを記憶し、データを転送する度にこのアドレスは、インクリメンタ35によって増加される。32はデータ転送ワード数レジスタであり、転送するワード数を記憶しており、1回の転送毎にデクリメンタ36によって転送ワード数が減少される。そして転送ワード数がゼロになった時がそのディスクリプタの転送動作の終了である。

【0028】33はチェーン先アドレスレジスタであり、ディスクリプタ情報のチェーン先アドレスを記憶するが、アレイチェーン式のディスクリプタ領域の場合には、連続した次のディスクリプタ情報のアドレスを計算したデータが記憶される。34はフォーマット指定レジスタであり、最初はCPUから入力されたフォーマット指定情報が書き込まれるが、以後はディスクリプタ情報を読み込む毎にそのフォーマット指定情報が記憶される。ディスクリプタ読み込み制御手段30は、このフォーマット指定情報に基づいて、ディスクリプタ情報の読み込みを行なう。

【0029】DMA転送動作は、図4に示すように、まずステップ401でCPUがメモリへ図2に示すようなディスクリプタ情報を書き込む。DMA転送の必要が生じると、ステップ402で、CPUが最初のディスクリプタ情報の開始アドレスとそのフォーマットをDMACに書き込む。DMACでは、これに応じて、ステップ411でアドレスがチェーン先アドレスレジスタ33に書き込まれ、フォーマット指定情報がフォーマット指定レジスタ34に書き込まれる。そしてステップ403で、CPUがDMACに起動信号を出力する。

【0030】DMACは、ステップ413、414及び420で、フォーマット指定レジスタ34に記憶されたフォーマットに従って、チェーン先アドレスレジスタ33のアドレスから所定量のディスクリプタ情報を読み込み、データ転送アドレスの部分をデータ転送アドレスレジスタ31に記憶し、転送ワード数をデータ転送ワード数レジスタ32に記憶する。この時同時にフォーマット指定情報も読み込まれ、フォーマット指定レジスタ34に記憶される。

【0031】ステップ415と416では、データ転送アドレスレジスタ31のアドレスからデータ転送を開始し、データ転送ワード数レジスタ32に記憶されたワード数分転送を行なう。次にステップ417で、最終のデ

ィスクリプタであるかを判定する。終了する場合は、DMACは待機状態に戻る。

【0032】最終ディスクリプタでなければ、ステップ418で、読み込んだディスクリプタ情報に基づいて次のディスクリプタ情報のフォーマットがアレイチェーンモードかリンクアレイチェーンモードかを判定する。もしアレイチェーンモードであれば、ステップ420でディスクリプタ情報のアドレスを増加させ、リンクアレイチェーンモードであれば、ステップ419でディスクリプタ情報のチェーン先アドレスの部分を読み込み、チェーン先アドレスレジスタ33に設定する。

【0033】そして再びステップ413に戻り、終了するまでステップ413から420を繰り返す。例えば、図2に示すようなディスクリプタ情報であれば、ステップ411でフォーマット指定レジスタ34にディスクリプタ領域1がフォーマット1であることが設定され、ステップ413でディスクリプタ領域1の読み込みが行なわれる。ディスクリプタ領域1には次のディスクリプタ領域2がフォーマット1であることを示すフォーマット指定情報が含まれている。ディスクリプタ領域1に指示されたDMAデータ転送を行った後、フォーマット指定レジスタ34のフォーマットに従ってディスクリプタ領域2の読み込みを行なう。これをディスクリプタ領域n-1まで繰り返す。

【0034】ディスクリプタ領域n-1のフォーマット指定情報は、ディスクリプタ領域nがフォーマット2であることを示している。従ってディスクリプタ領域nはフォーマット2としてチェーン先アドレスも読み込まれる。このチェーン先アドレスはディスクリプタ領域1のアドレスを示しており、最初に戻ることになる。このようにしてループ状のチェーン式DMA転送が行なえる。

【0035】なお、ディスクリプタ情報はフォーマット指定情報を含むため、従来のリンクアレイチェーン式に比べて利点がないように見えるが、フォーマットが2種類であればフォーマット指定情報は1ビットで表わせ、データ量のはるかに少ない。もちろんフォーマットは2種とは限らず種類の数を増加でき、その場合でもデータ量はあまり多くなならない。例えば、256種類のフォーマットを選択するとしても1バイトで収まる。このことにより、ループ状のチェーンを構成する場合、従来のリンクアレイチェーン方式で実施するよりも、ディスクリプタ領域が削減されると共に、チェーン先アドレスを読み込むバスサイクル数が削減される。

【0036】更にこれまで説明した態様では、フォーマット指定情報は次のディスクリプタのフォーマットを示したが、フォーマット指定情報がそのディスクリプタ自体のフォーマットを示し、まずそのフォーマットを含む共通部分の読み込みを行なった後、そのフォーマットに従って残りのディスクリプタを読み込むようにしてもよい。この場合は、最初にCPUがDMACにフォーマッ

トを書き込む動作が必要なくなる。

【0037】

【実施例】図5と図6は実施例におけるDMAコントローラの構成図であり、構成を2図に分割して示してある。図において、50はディスクリプタ読み込み制御部であり、501はコントロールレジスタであり、502はバス制御回路である。511と512は転送アドレス用レジスタであり、転送アドレスのLowバイトとHighバイトを保持する。521と522は転送バイト数のLowバイトとHighバイトを保持するレジスタである。なおここでは転送データ量をバイト単位で表わす。531と532は次のディスクリプタのアドレスのLowバイトとHighバイトを保持するディスクリプタポインタレジスタである。54はフォーマット指定レジスタである。以上のコントロールレジスタ501、転送アドレスレジスタ511、512、転送バイト数レジスタ521、522、ディスクリプタポインタレジスタ531、532、及びフォーマット指定レジスタ54は、8ビットのバス1に接続され、このバス1は外部に接続されている。

【0038】転送アドレスレジスタ511、512及びディスクリプタポインタレジスタ531、532はバス2にも接続されており、このバス2にはインクリメンタ55とアドレスラッチ513が接続されている。このバス2により次のディスクリプタアドレスの変更が行われ、アドレスラッチ513からシステム側のアドレスバスにDMA転送時のアドレスが出力される。

【0039】転送バイト数レジスタ521、522はバス3にも接続され、バス3に接続されたデクリメンタ56によって転送したバイト数に応じた変化が行なわれる。次に図5、図6のDMACの動作を説明する。先ず、CPUによりDMAコントローラの初期設定と起動を行う。CPUはDMAコントローラのディスクリプタポインタレジスタLowバイトに第1番目に読み込むべきディスクリプタの先頭番地D1のLowバイト(D1)1の書き込みを行う。ディスクリプタポインタレジスタLowバイトへの設定値(D1)は、バス1を経由し、リード/ライト制御ブロックから発生する信号c(バス1→ディスクリプタポインタLowバイト)パルスによって、ディスクリプタポインタLowバイトに書き込まれる。

【0040】CPUはDMAコントローラのディスクリプタポインタレジスタHighバイトに第1番目に読み込むべきディスクリプタの先頭番地D1のHighバイト(D1)の書き込みを行う。ディスクリプタポインタレジスタHighバイトへの設定値(D1)は、バス1を経由し、リード/ライト制御ブロックから発生する信号d(バス1→ディスクリプタポインタHighバイト)パルスによって、ディスクリプタポインタHighバイトに書き込まれる。

【0041】CPUはDMAコントローラのフォーマッ

ト指定レジスタに第1番目に読み込むべきディスクリプタのフォーマットを書き込む。第1番目に読み込むべきディスクリプタのフォーマットを示すデータは、バス1を経由し、リード/ライト制御ブロックから発生する信号b(バス1→フォーマット指定レジスタ)パルスによって、フォーマット指定レジスタに書き込まれる。

【0042】CPUはDMAコントローラのコントロールレジスタにDMA起動コマンドを書き込む。DMA起動コマンドデータは、バス1を経由し、リード/ライト制御ブロックが発生するa信号(バス1→コントロールレジスタ)パルスによって、コントロールレジスタに書き込まれる。次にディスクリプタの読み込み時の動作を説明する。

【0043】図7は、フォーマット1のディスクリプタ読み込みのタイミングチャートである。コントロールレジスタへのDMA起動コマンドの書き込みまたはDMAデータ転送の終了(ディスクリプタに指定されたバイト数分のDMAデータ転送が終了)により、ディスクリプタ読み込み制御部はディスクリプタ転送要求信号を

“H”にする。バス制御回路は、ディスクリプタ転送要求信号が“H”になると、CPUに対してホールドを要求する。ホールド許可が入力されると、バス制御回路はアドレス出力、リード出力、ライト出力をHi-Z状態から出力状態にし、リードバスサイクルを発生する。ここでは、バス制御回路の詳細は割愛する。リードバスサイクルで出力するアドレスは、信号iによってバス2に出力されたディスクリプタポインタLow、Highレジスタの内容を信号oの立ち上がりエッジでアドレスラッチブロックに記憶した内容である。この時のバス2の内容は、信号nの立ち上がりエッジでインクリメンタブロックにも取り込まれる。インクリメンタブロックがインクリメントした結果は、次のクロックサイクルで信号kによってバス2に出力され、信号1の立ち上がりエッジでディスクリプタポインタLow、Highレジスタに格納される。一方、リードバスサイクルでバス1に入力されるデータは、ディスクリプタ読み込み制御部が順次発生する信号e、f、g、h、bのパルスで、各レジスタに取り込まれる。信号bのパルスで取り込まれたフォーマット指定レジスタの内容は、次のディスクリプタ読み込み時のフォーマット示す。また、ディスクリプタポインタLow、Highレジスタには、今読み込んだディスクリプタの直後のアドレスが残る。

【0044】図8は、フォーマット2のディスクリプタ読み込みのタイミングチャートである。コントロールレジスタへのDMA起動コマンドの書き込みまたはDMAデータ転送の終了(ディスクリプタに指定されたバイト数分のDMAデータ転送が終了)により、ディスクリプタ読み込み制御部はディスクリプタ転送要求信号を

“H”にする。バス制御回路は、ディスクリプタ転送要求信号が“H”になると、CPUに対してホールドを要

求する。ホールド許可が入力されると、バス制御回路はアドレス出力、リード出力、ライト出力をHi-Z状態から出力状態にし、リードバスサイクルを発生する。ここでは、バス制御回路の詳細は割愛する。リードバスサイクルで出力するアドレスは、信号iまたは信号kによってバス2に出力されたディスクリプタポインタLow, High レジスタの内容またはインクリメンタブロックに残っていた内容を信号oの立ち上がりエッジでアドレスラッチブロックに記憶した内容である。この時のバス2の内容は、信号nの立ち上がりエッジでインクリメンタ

ブロックにも取り込まれる。一方、リードバスサイクルでバス1に入力されるデータは、ディスクリプタ読み込み制御部が順次発生する信号e, f, g, h, c, d, bのパルスで、各レジスタに取り込まれる。信号bのパルスで取り込まれたフォーマット指定レジスタの内容は、次のディスクリプタ読み込み時のフォーマットを示す。また、ディスクリプタポインタLow, High レジスタには、信号c, dのパルスで取り込まれた値が残る。

【0045】DMAデータ転送に関する説明は省略するが、DMAデータ転送のタイミングチャートのみを図9に示す。図7及び図8のディスクリプタの読み込みは、コントロールレジスタ501へのDMA起動コマンドの書き込みが行なわれた時、又は転送バイト数レジスタの内容がゼロになった時かつ最終のディスクリプタを読みこんでいない時に行なわれる。

【0046】なお上記実施例においては、フォーマット指定情報として1バイトを割り当て256種類のフォーマットを選択可能にしたが、フォーマットの種類が上記のフォーマット1とフォーマット2の2種類だけであれば、フォーマット指定情報には1ビットを割り当てればよい。そこで図10のディスクリプタ情報の他の構成例に示すように、転送ワード数を表わす部分の1ビットをフォーマット指定ビットとすれば、ディスクリプタ領域の大きさは従来通りであり、ディスクリプタの読み込み動作に要する動作も増加しない。但し、表現できる転送ワード数が半分になる。しかしチェーン式のDMA転送では一度に転送するデータ量があまり大きくなることはないため実用上は問題ない。

【0047】これまでの説明はすべてディスクリプタ情報が転送先アドレスと転送データ数、更にフォーマット

この部分と上記のSC部分はブロックに対するデータ転送の終了時にDMACからDMA転送で書き出される。最後の3バイトはチェーン先アドレスを示す部分である。

【0048】通常のDMACでは、データ転送時点で状態情報を書き出すが、この情報がフォーマットによって異なることがある。その場合には、フォーマット指定レジスタの内容に応じた書き出しを行なう必要があるのはいうまでもない。これまでの説明では、フォーマット指定情報は次のディスクリプタ情報の形式を示していたが、最初に読み込む情報にフォーマット指定情報が含まれるならば、フォーマット指定情報がそのディスクリプタ情報の形式を示し、読み込んだフォーマット指定情報に応じて残りの部分の読み込みを行なうことが可能になる。

【0049】図12は、図11のディスクリプタ情報の構成で、フォーマット指定情報がそのディスクリプタ情報のフォーマットを示しているとした時のDMA転送動作のフローチャートである。図12の説明を行なうが、従来と共通な部分の説明は省略し、異なる点のみを説明する。ステップ201ではディスクリプタ情報の開始アドレスのみをDMACに書き込み、フォーマット指定情報は書き込まない。そして起動後、DMACでは、ステップ204で図11の1の部分を読み込む。ステップ205でSC部分の終了情報を判定する。終了でなければ、ステップ206でフォーマット1であるかフォーマット2であるかを判定する。フォーマット1であれば、ステップ207でディスクリプタアドレスを増加させ、ステップ210に進む。フォーマット2であれば、ステップ208で図11の3の部分、すなわちチェーン先アドレスを読み込み、ステップ209でそれをディスクリプタアドレスに設定する。ステップ210のデータ転送後、ステップ211で図11の2の部分を書き出す。以降、終了するまでステップ204から211を繰り返す。

【0050】

【発明の効果】以上説明した様に、本発明によればディスクリプタ領域の削減とバスサイクル数の削減ができる。特に、近年の一般的なシステムではバスネックになっているものが多く、バスサイクル数の削減は、様々な装置の性能向上に寄与するところが多い。

【図面の簡単な説明】

【図1】本発明におけるディスクリプタ情報の基本構成例を示す図である。

【図2】本発明におけるディスクリプタ全体構成例を示す図である。

【図3】本発明のDMAコントローラの基本構成図である。

【図4】本発明における基本的なDMA転送動作のフローチャートである。

【図5】実施例におけるDMAコントローラの構成の一部を示す図である。

【図6】図5の残り部分を示す図である。

【図7】実施例におけるフォーマット1のディスクリプタの読み込みのタイミングチャートである。

【図8】実施例におけるフォーマット2のディスクリプタの読み込みのタイミングチャートである。

【図9】実施例におけるDMAデータ転送のタイミングチャートである。

【図10】ディスクリプタ情報の他の構成例を示す図である。

【図11】ディスクリプタ情報の詳細な構成例を示す図である。

【図12】第2実施例におけるDMA転送動作のフローチャートである。

【図13】従来のアレイチェーンモード設定時のDMA

転送動作を示すフローチャートである。

【図14】従来のアレイチェーンモードのディスクリプタ構成を示す図である。

【図15】従来のリンクアレイチェーンモード設定時のDMA転送動作を示すフローチャートである。

【図16】従来のリンクアレイチェーンモードのディスクリプタ構成を示す図である。

【図17】従来のDMAコントローラのブロック構成図である。

【符号の説明】

30…ディスクリプタ情報読み込み制御手段

31…転送アドレスレジスタ

32…転送データ量レジスタ

33…チェーン先アドレスレジスタ（ポインタアドレスレジスタ）

34…フォーマット指定レジスタ

【図1】

【図10】

本発明におけるディスクリプタ情報の基本構成例

データ転送アドレス
データ転送ワード数
フォーマット指定

フォーマット1

データ転送アドレス
データ転送ワード数
チェーン先アドレス
フォーマット指定

フォーマット2

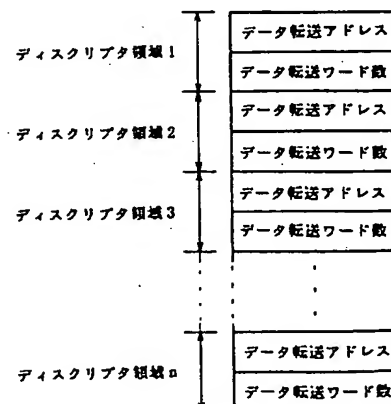
データ転送アドレス
F 転送ワード数

データ転送アドレス
チェーン先アドレス
F 転送ワード数

F：フォーマット指定ビット

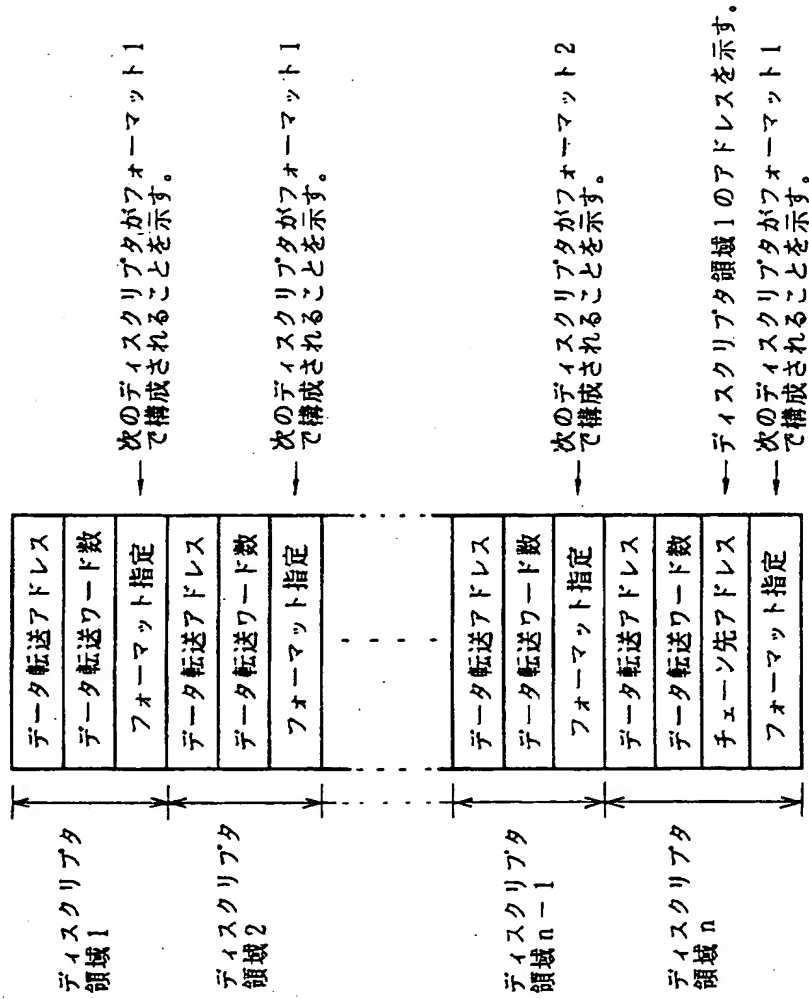
【図14】

アレイチェーンモード時のディスクリプタ構成



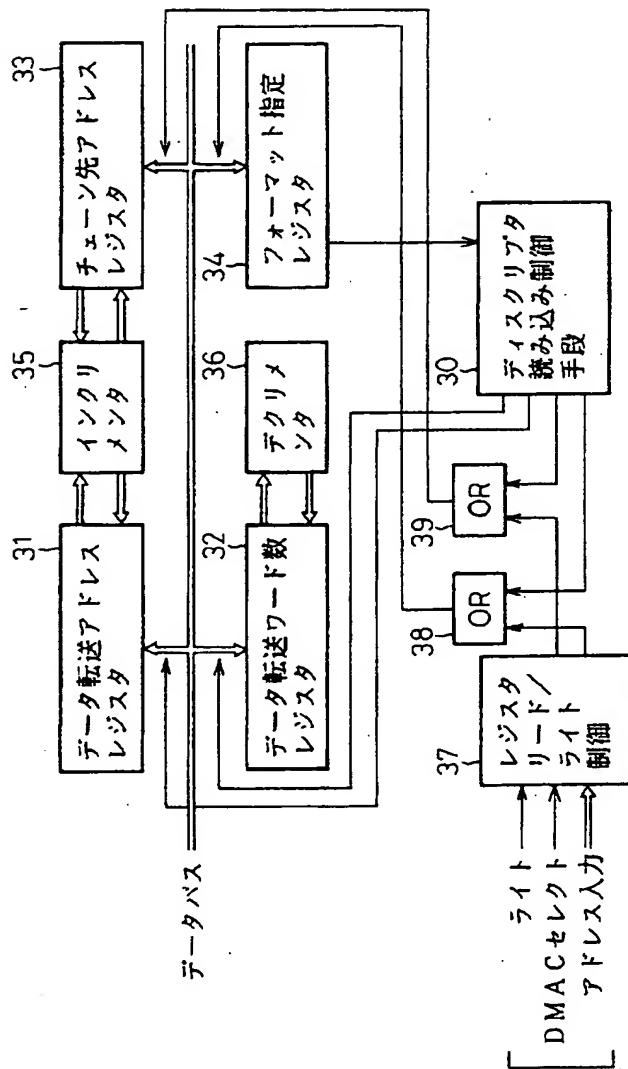
【図2】

本発明におけるディスクリプタ全体構成例

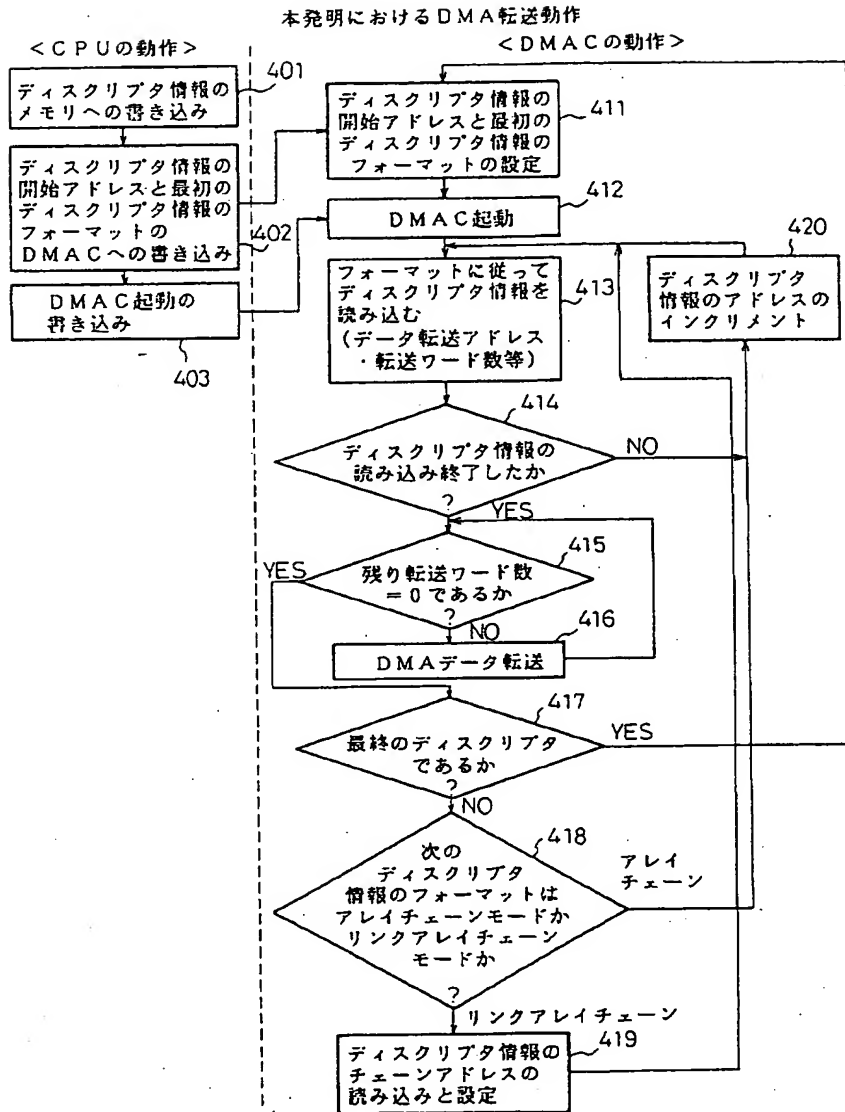


【図3】

本発明のDMAコントローラの基本構成

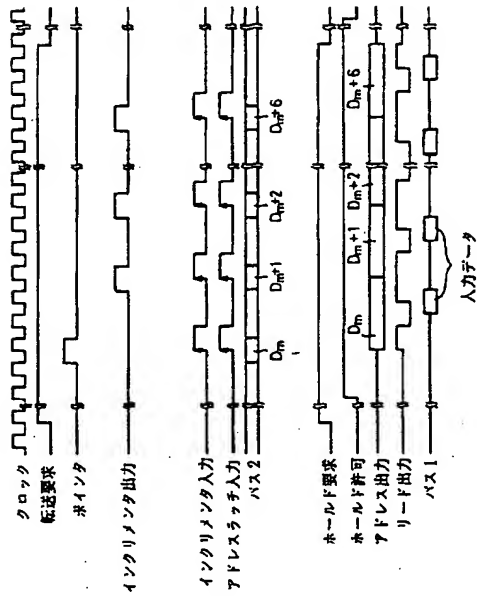


【図4】



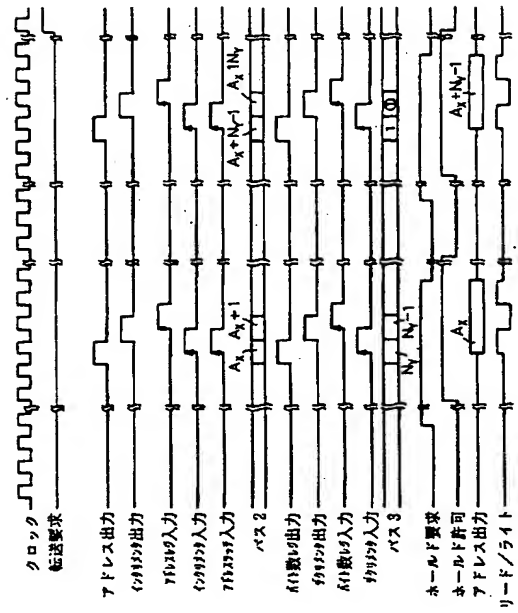
【図8】

実施例におけるフォーマット2のディスクリプタの読み込み時のタイミングチャート



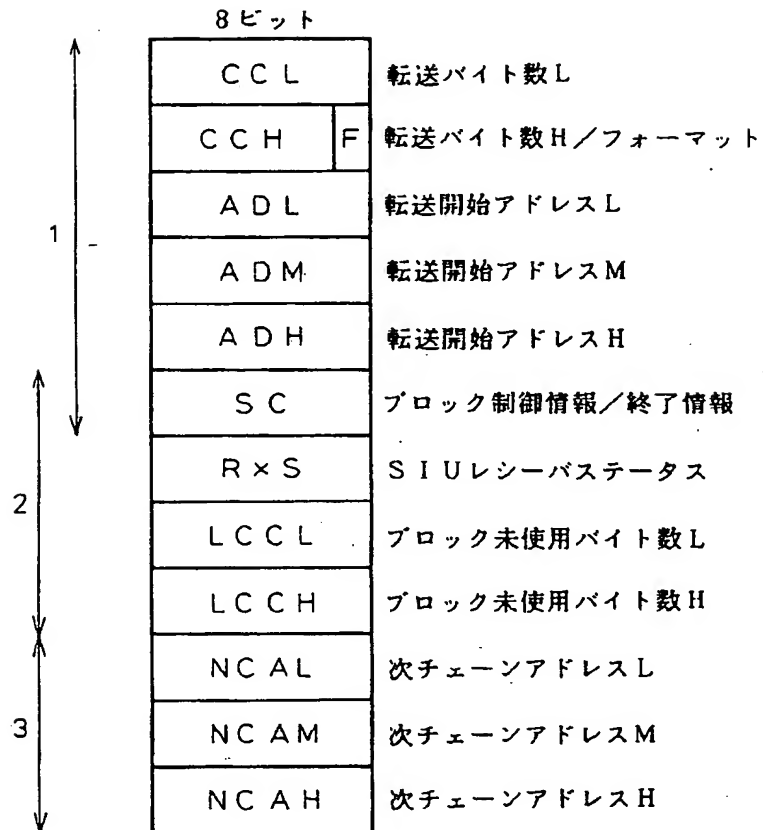
【図9】

実施例におけるDMAデータ転送のタイミングチャート



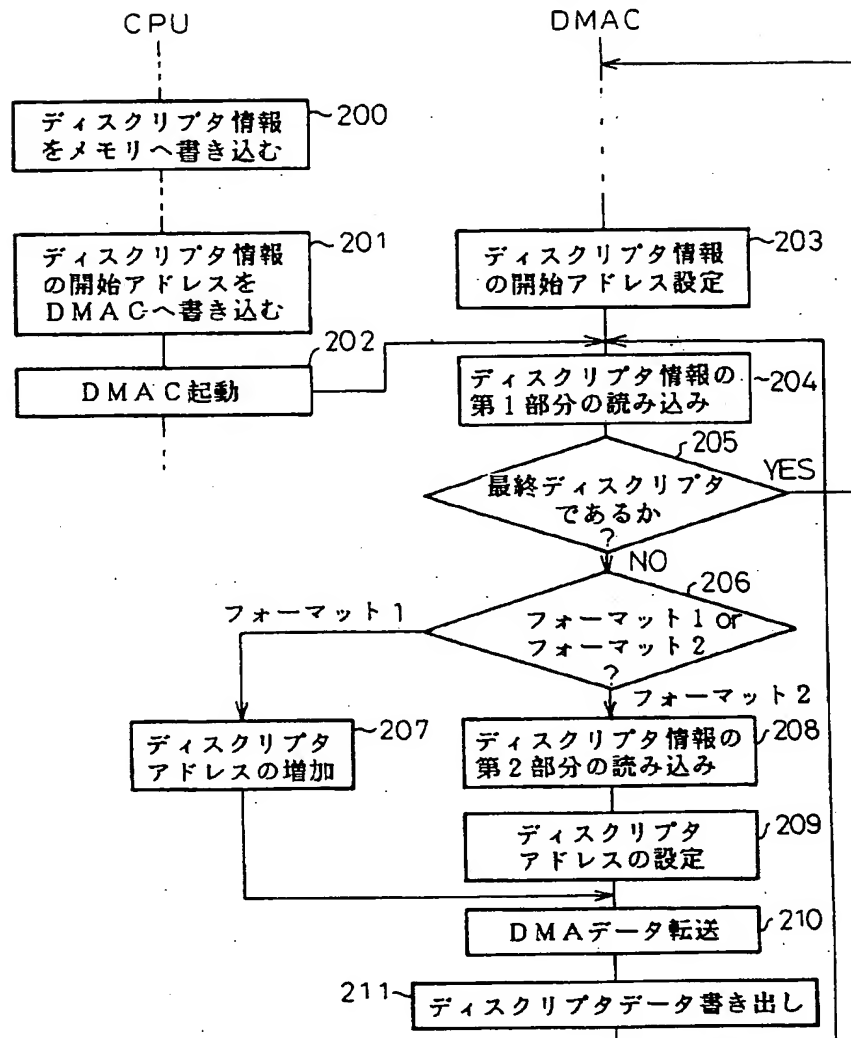
【図11】

ディスクリプタ情報の構成例



【図12】

第2実施例におけるDMA転送動作



【図13】

従来のDMACによるDMA転送動作（アレイチェーンモード設定時）
 <CPUの動作>

ディスクリプタ情報の
メモリへの
書き込み

1301

ディスクリプタ情報の
開始アドレスと
アレイチェーンフォーマット
のDMACへの
書き込み

1302

DMAC起動
の書き込み

1303

<DMACの動作>

ディスクリプタ情報の
開始アドレスとアレイ
チェーンフォーマット
の設定

1311

DMAC起動

1312

フォーマットに従って
ディスクリプタ情報を
読み込む
(データ転送アドレス
・転送ワード数等)

1313

ディスクリプタ
情報のアドレスの
インクリメント

1318

ディスクリプタ情報の
読み込み終了したか

1314

YES

NO

残り転送ワード数
= 0 であるか

1315

YES

NO

DMAデータ転送

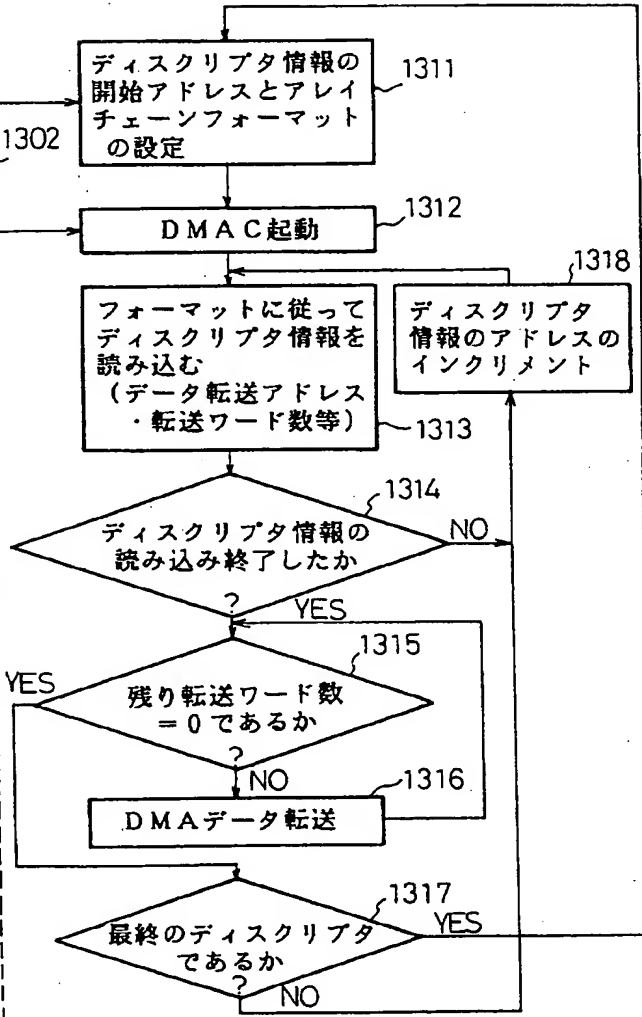
1316

最終のディスクリプタ
であるか

1317

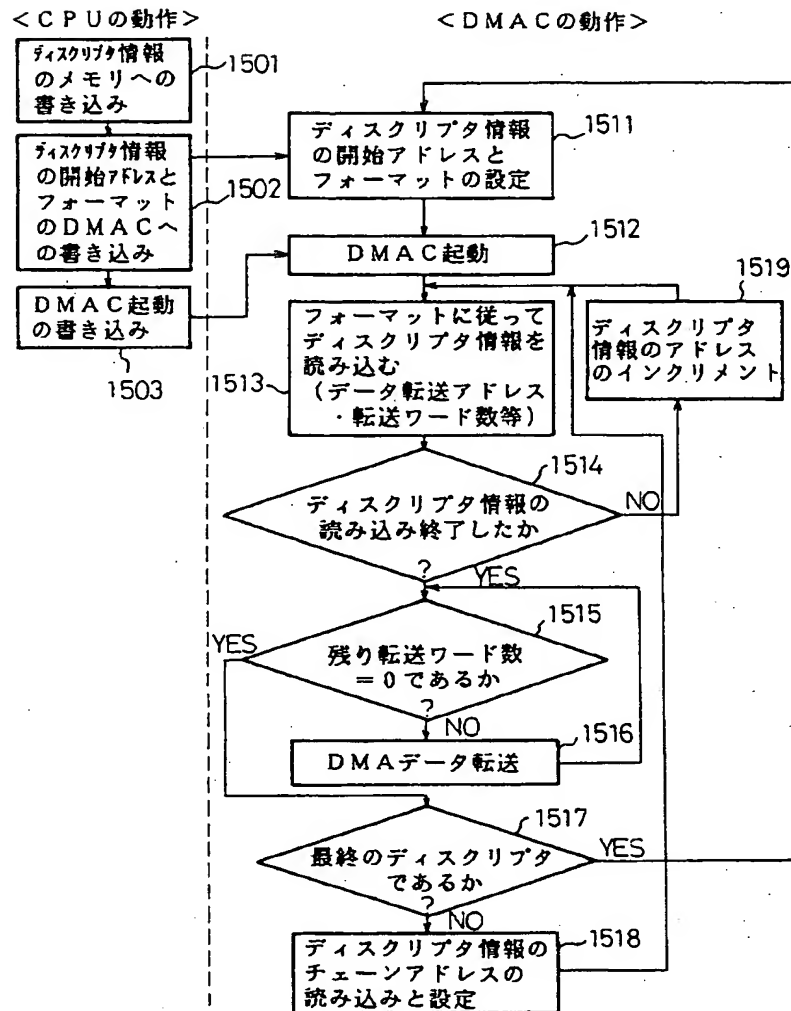
YES

NO



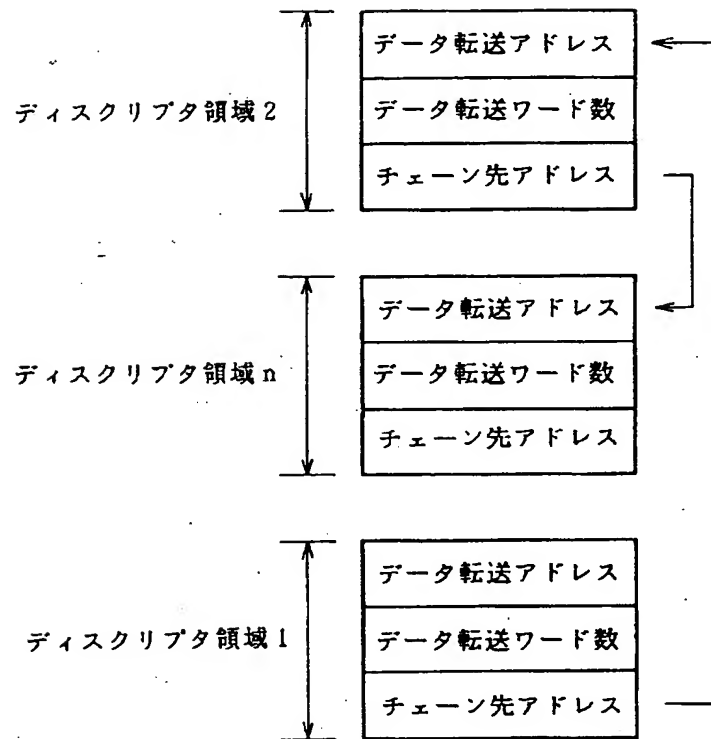
【図15】

従来のDMACによるDMA転送動作（リンクアレイ
チェーンモード設定時）



〔図16〕

リンクアレイチェーンモード時のディスクリプタ構成



【図17】

従来のDMAコントローラのブロック構成図

